ConneXions™

The Interoperability Report

June 1989 Volume 3, No. 6

ConneXions— The Interoperability Report tracks current and emerging standards and technologies within the computer and communications industry.

In this issue:

ConneXions is published by Advanced Computing Environments, 480 San Antonio Road, Suite 100, Mountain View, CA 94040, USA. Phone: 415-941-3399.

ISSN 0894-5926

From the Editor

Following last month's description of the X.400 Message Handling System, we continue our series *Components of OSI* with an overview the X.500 Directory Services. Unlike most of the directory systems in use today X.500 will support not only electronic messaging, but all of the OSI application layer systems. The article is by Steve Benford of the University of Nottingham.

User support and information services are important factors in the daily operation of any computer network. An example of such a support operation is the Merit/NSFNET Information Services. Laura Kelleher of Merit describes the system on page 10.

Network management continues to be an important issue for both the vendor and research communities. In April, the Internet Activities Board (IAB) issued RFC 1095 describing the CMOT architecture and re-issued RFC 1067 (as RFC 1098) which specifies the SNMP protocol. This action by the IAB effectively made CMOT and SNMP equal players in the TCP/IP network management world. The full text of these announcements is given on page 15.

The Design and Implementation of the 4.3BSD UNIX Operating System is an important book for anyone involved with UNIX development and TCP/IP networking. We asked Barry Shein to review this newly released text. His comments appear on page 16.

In our August 1988 issue we described the DCA Protocol Testing Laboratory. The system has since become the basis for all testing of TCP/IP protocols for the DoD as part of the the National Voluntary Laboratory Accreditation Program (NVLAP). Martin Gross of the Defense Communications Engineering Center (DCEC) gives the current status on page 18.

While it is the policy of *ConneXions* to encourage quotation with attribution, we did not expect to see what appeared in *PC WEEK\CONNECTIVITY* on April 3rd, 1989. Under the heading "Behind the Scenes at Interop: Network Disaster" the writer, David Strom, has concocted a bizarre mix of extracts from our February 1989 article about the INTEROP™ network, and his own opinions on how to build and install a show network. The result is a piece so full of misleading and incorrect information that it would take more than a mere paragraph to rebut. Suffice it to say that we are not entirely pleased with this kind of sloppy journalism. I encourage you to read the original *ConneXions* article, and compare it with Mr. Strom's piece and then draw your own conclusions.

Components of OSI: The OSI Directory Service

by Steve Benford, University of Nottingham

Introduction

This article provides an overview of the *OSI Directory Service* as specified in the joint ISO/CCITT international standard ISO IS 9594, CCITT X.500 [1]. This standard has been jointly specified by the ISO and CCITT standardisation bodies to meet the urgent requirement for Directory services within the expanding OSI environment. The need for a Directory to support OSI applications has been apparent for several years. In particular, the requirement for a Directory to support X.400 Message Handling Systems has provided a major driving force behind the development the OSI Directory. However, the Directory is intended to support a wider range of applications than just electronic mail.

The article takes a brief tour through the OSI Directory standard, outlining its purpose and describing its key features. In particular, it is aimed at readers from the internet community who, although not cognizant of X.500 itself, may be familiar with existing name-servers such as the Internet *Domain Name System*. (DNS). The emphasis of the article is on abstract directory models and general functionality as opposed to details of distribution. These details may be found in the OSI Directory standard, if required.

Scope of X.500

The broad aim of the OSI Directory is to specify a standard for the inter-connection of Directory Services to provide a *Global Directory*, supporting network users and applications with the information required for communication. More specifically, Directory Services maintain information describing objects such as humans, organisations, application entities, distribution lists and network hardware. A key feature of the Global Directory is the provision of a global name space for these objects, under which the responsibility for naming is distributed.

By supporting applications in the distributed naming of objects, the Directory is fulfilling the traditional role of a "nameserver." Readers may be familiar with several existing nameservers such as the DNS [2] and the Clearinghouse [3]. However, it is important to realise that the scope of Directory Services extends beyond that of nameservers. In addition to the resolution of names and support for applications such as electronic mail, the Directory acts as a general information service, perhaps interacting directly with humans. In this sense it is also analogous to the white and yellow page telephone directories of today, although these are primitive in comparison. One role of the Directory is to provide people with communication information. This is supported by rich functionality for searching and browsing information. In general, the Global Directory can be expected to support a wider range of applications and users than existing nameservers. Consequently, it contains a larger volume of more diverse information and therefore requires broader functionality.

Structure

The X.500 standard defines those aspects of Directory Services necessary for interconnection. The standard is divided into 8 parts describing such issues as the abstract structure of Directory information, naming, the abstract service definition and procedures for distributed operation. These parts are listed below along with their respective ISO and CCITT references (noted as [CCITT ref/ISO ref]).

- Part 1: Overview of Concepts, Models, and Service [X.500/9594-1]: Provides a short overview of the Directory, its information, possible operations and distributed organisation.

- Part 2: Models [X.501/9594-2]: Describes a number of basic Directory models including the naming model, abstract information model and functional model.

- Part 3: Abstract Service Definition [X.511/9594-3]: Defines the service offered to Directory users in terms of abstract operations forming a standard *Directory Access Protocol*.

- Part 4: Procedures for Distributed Operation [X.518/9594-4]: Specifies the distributed realisation of the directory in terms of cooperation between a number of autonomous servers called *Directory System Agents*.

- Part 5: Protocol Specifications [X.519/9594-5]: Defines Directory protocols in relation to the OSI model.

- Part 6: Selected Attribute Types [X.520/9594-6] and Part 7: Selected Object Classes [X.521/9594-7]: Define some standard Directory information types representing well known types of object.

- Part 8: Authentication Framework [X.509/9594-8]: Describes the role of the Directory service in providing itself and other applications with a framework supporting "simple" and "strong" authentication.

The remainder of this article considers the major features of the OSI Directory in greater detail.

Information model

The Directory "information model" specifies the abstract structure of directory information. Each object, known to the Directory, is represented by an *entry*. The set of all entries is called the *Directory Information Base*. Each entry consists of a set of *attributes* representing specific known facts about the object. For example, an attribute might represent a mail address or a member of a distribution list. Each attribute has an *attribute type*, indicating the type of information represented, and a *value*, containing the information. An entry may contain more than one attribute of a given type. Attribute types are globally unique, being represented by *Object Identifiers* as defined within the Abstract Syntax Notation 1 (ASN.1) [4] standard. However, character strings are used in documentation for reasons of legibility.

An object identifier is a unique hierarchical sequence of numbers, allocated by an administrative organisation such as a standards body, uniquely identifying an object (e.g. an attribute type). For example, the attribute type "commonName" is identified by the object identifier "5.1.5.3" (5 = joint-ISO-CCITT, 1 = modules, 5 = selectedAttributeTypes, 3 = commonName).

For those familiar with the Internet Domain Name System (DNS), entries and attributes play a similar role to "resources" and "resource records." However, unlike the DNS, the Directory will support a wide range of attribute types relevant to many different applications.

continued on next page

The OSI Directory Service *(continued)*

Entries are grouped into generic *object classes* based on the type of object they represent (e.g. *organisational person* or *group of names*). Each entry contains a special attribute indicating to which object class it belongs.

Naming

Naming is critical to the operation of the Directory service. The above definition of the Directory Information Base might suggest a relational style information model. However, in order to support the distributed management of global names for objects, entries are arranged into a hierarchical structure called the *Directory Information Tree* (DIT). Each vertex of the DIT is an entry, labeled with a *Relative Distinguished Name*, unique among its siblings. The relative distinguished name (RDN) is composed of a subset of the entry's attributes called *distinguished* attributes. Distributed name management is achieved by assigning the responsibility for choosing each entry's RDN to its parent in the DIT. The parent is known as the entry's *Naming Authority*.

Each entry has a globally unique and unambiguous *Distinguished Name*, composed of the ordered sequence of RDNs encountered on the path from the root of the DIT to the entry. Distinguished names provide the basic handle on entries and their contents. Relative distinguished names, and hence distinguished names, are generally chosen to be stable over long periods of time and user friendly. A distinguished name need not be the only name for an entry. An alternative name or alias may be supported by the use of special pointer entries called *Alias Entries*. Alias entries do not contain any other attributes beyond their distinguished attributes. Furthermore, they may only be leaf entries of the DIT.

A Directory user identifies an entry by supplying an ordered set of purported attribute value assertions (i.e. type = value pairs) forming a *Purported Name*. The purported name is mapped onto the desired entry by the process of *name verification* which performs a distributed tree walk through the DIT, dereferencing any aliases which are encountered. Once again, there is a strong similarity between the OSI Directory naming model and the DNS naming model. Figure 1 shows an example DIT.

Abstract Service Definition

The Directory *Abstract Service Definition* specifies the functionality of the Directory in terms of a set of abstract ports and operations forming the *Directory Access Protocol*. Broadly speaking, directory functionality can be divided into three categories, encapsulated by the "Read," "Search" and "Modify" ports respectively.

The Read port supports the retrieval of information from specific named entries. This allows a general name to attributes mapping, analogous to the white pages telephone directory. The Read port identifies the following three operations:

- The "Read" operation returns the values of specified attributes from a single named entry.
- The "Compare" operation returns an indication of whether a named entry contains a specified attribute type and value.
- The "Abandon" operation allows the termination of Directory operations where possible.

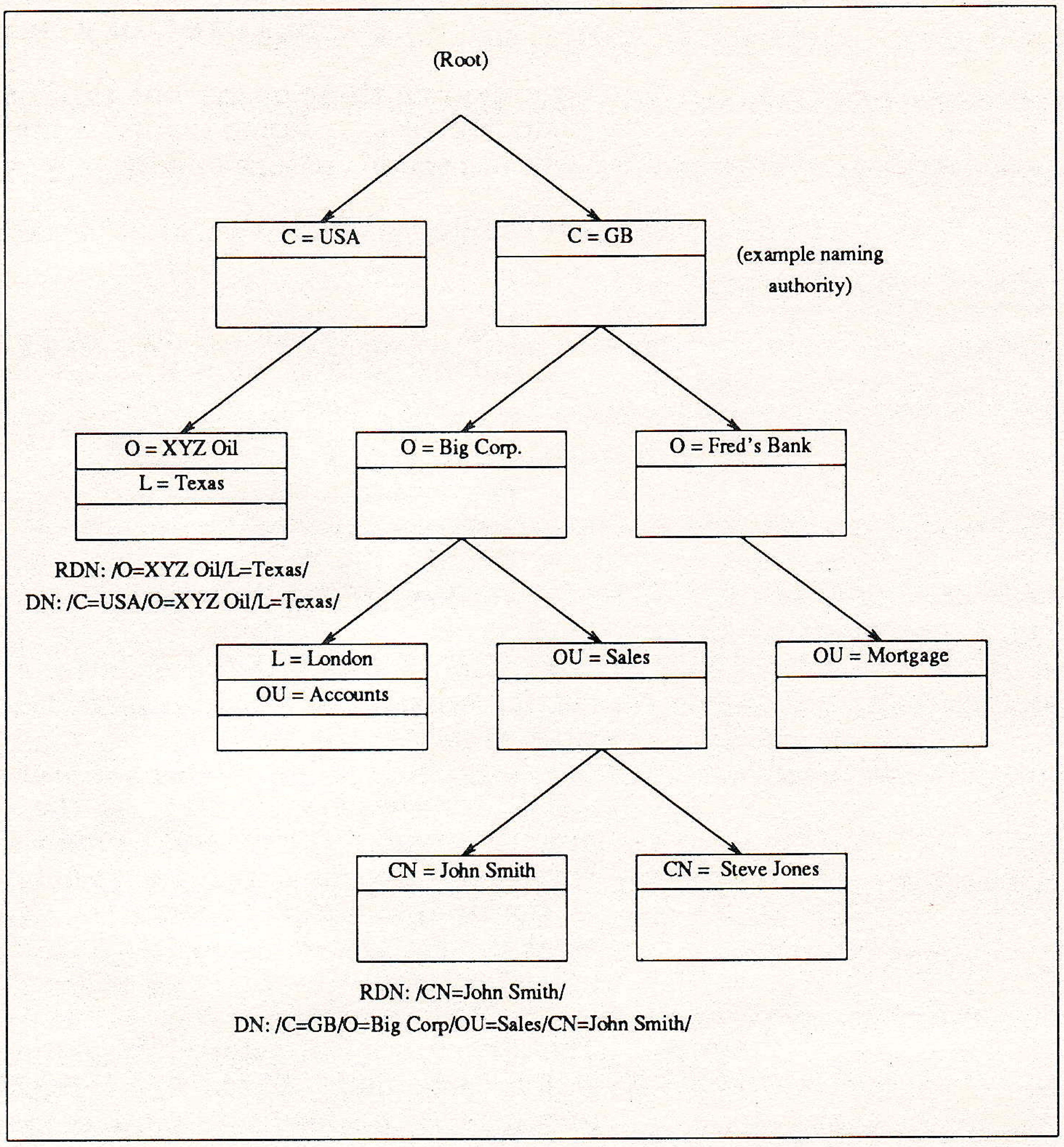


Figure 1: Example Directory Information Tree

The general browsing of information is achieved via the Search port, supporting two operations:

- The "List" operation returns the names of the children of a named entry and is used for browsing the DIT structure.

- The "Search" operation supports the searching of DIT subtrees for entries matching specific patterns of attributes. The user names a subtree of the DIT, specifies some target attribute types and formulates an expression combining a number of attributes using logical "and", "or" and "not" operators. The expression is called a *filter*. The operation returns the values of the target attributes from those entries in the named subtree, matching the filter. For example, to obtain the telephone numbers of all programmers called Smith working for Big Corp, a user could request a search on the subtree "C = GB, O = Big Corp" with the filter "description = programmer AND surname = Smith" requesting attributes of type "telephone number."

The functionality of the Search port is required to support human interaction with the Directory and is analogous to that of the "yellow pages" telephone directory.

continued on next page

The OSI Directory Service *(continued)*

By providing these operations the Directory is extending beyond the basic nameserver functionality of many existing systems. Searching is an important feature of Directory Services.

The limited modification of information is achieved via the Modify port.

- The "Modify Entry" operation adds, replaces or removes a number of attributes within a named entry
- The "Add Entry" operation creates a new leaf entry in the DIT
- The "Remove Entry" operation deletes an entry from the DIT
- The "Modify Relative Distinguished Name" operation alters the RDN of a named leaf entry

It should be noted that the latter three operations only apply to entries which will remain as DIT leaves. They do not provide a general facility for building and manipulating the DIT.

Existing nameservers, such as the DNS, generally serve a small number of applications and contain a limited range of information. Directory Services support a variety of applications and the information they contain is necessarily diverse. In such an environment, mechanisms are required to specify and control the structure and use of information, otherwise chaos would ensue.

Schemas

The structure of directory information is governed by a set of rules called "schemas." These are integrity constraints ensuring that information conforms to well defined formats. Schemas specify rules for the following:

- The structure of names and hence the DIT
- The contents of entries in terms of attributes
- Possible attribute types
- Syntax for attribute values and rules for comparing them

For example, the Directory might include schemas defining the object class of entries called "person," with mandatory attributes "common name," "title" and "description." Whenever a "person" entry is created or modified, automatic integrity checking would ensure that these attributes are present. Furthermore, naming rules could guarantee that a person entry always has an "organisation" entry as a parent and that the values of its attributes are structured correctly (e.g. a "telephone number" is a sequence of numeric digits).

Each attribute is governed by a rule assigning it a unique object identifier and specifying the syntax of its value. In addition, this rule states the mechanism by which attributes of this type are compared with one another. Each entry in the DIT belongs to an "object class," governed by a schema. This schema specifies mandatory and optional attributes for entries of this class. Object Class definitions may be used to derive subclasses, supporting the inheritance and refinement of the attribute types defined for the super-class.

The ability to define subclasses is a powerful feature of the Directory. Naming rules govern which object classes may be children of which others in the DIT and therefore determine possible name forms.

The OSI Directory standard defines a number of standard attribute types and object classes. For example, the attribute types "common name" and "description," and the object classes "person" and "application process." Furthermore, applications may define their own schemas for application specific uses. The relationship between schemas and the Directory information model is shown below.

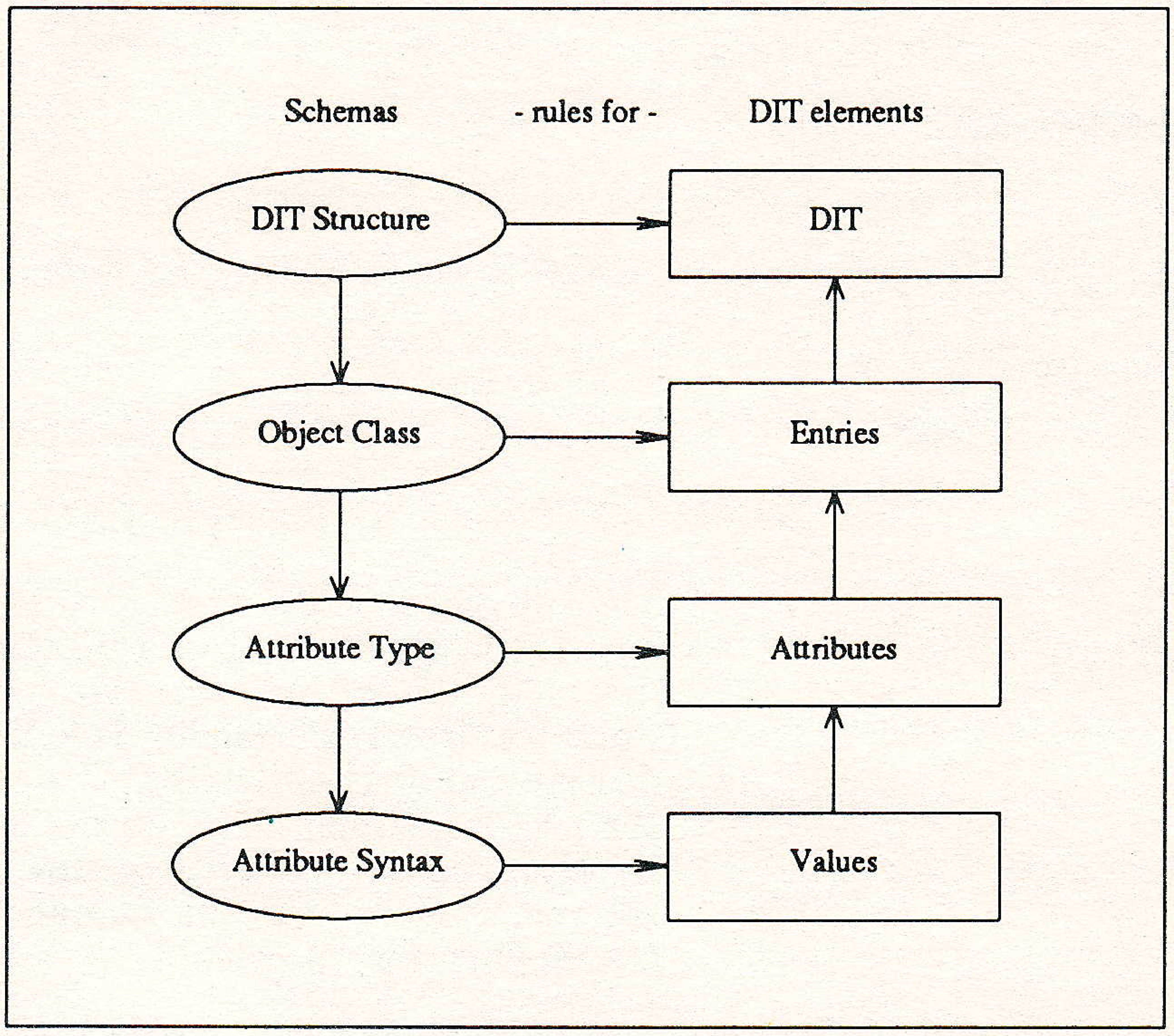


Figure 2: Schemas and their relationship to the Directory model

Distributed operation

The Global Directory will be a distributed service. Therefore, the X.500 standard specifies procedures for its distributed operation.

The information belonging to the Directory Information Base is shared between a number of application entities called *Directory System Agents* (DSAs). These cooperate to perform operations with each DSA knowing a fraction of the total Directory information. DSAs can be viewed as a combination of local database functionality and remote interface to the clients of users and other DSAs. DSAs may cooperate in order to execute operations. Cooperation may take several forms and requires the navigation of operations through the distributed system. To this end, the X.500 standard defines a DSA knowledge model and navigation algorithm. DSAs play a similar role to "nameservers" within the DNS.

A user accesses the Directory via an application entity called a *Directory User Agent* (DUA). DUAs manage associations with DSAs and present various interfaces to Directory users (human or application). They are similar in purpose to DNS "resolvers."

continued on next page

The OSI Directory Service *(continued)*

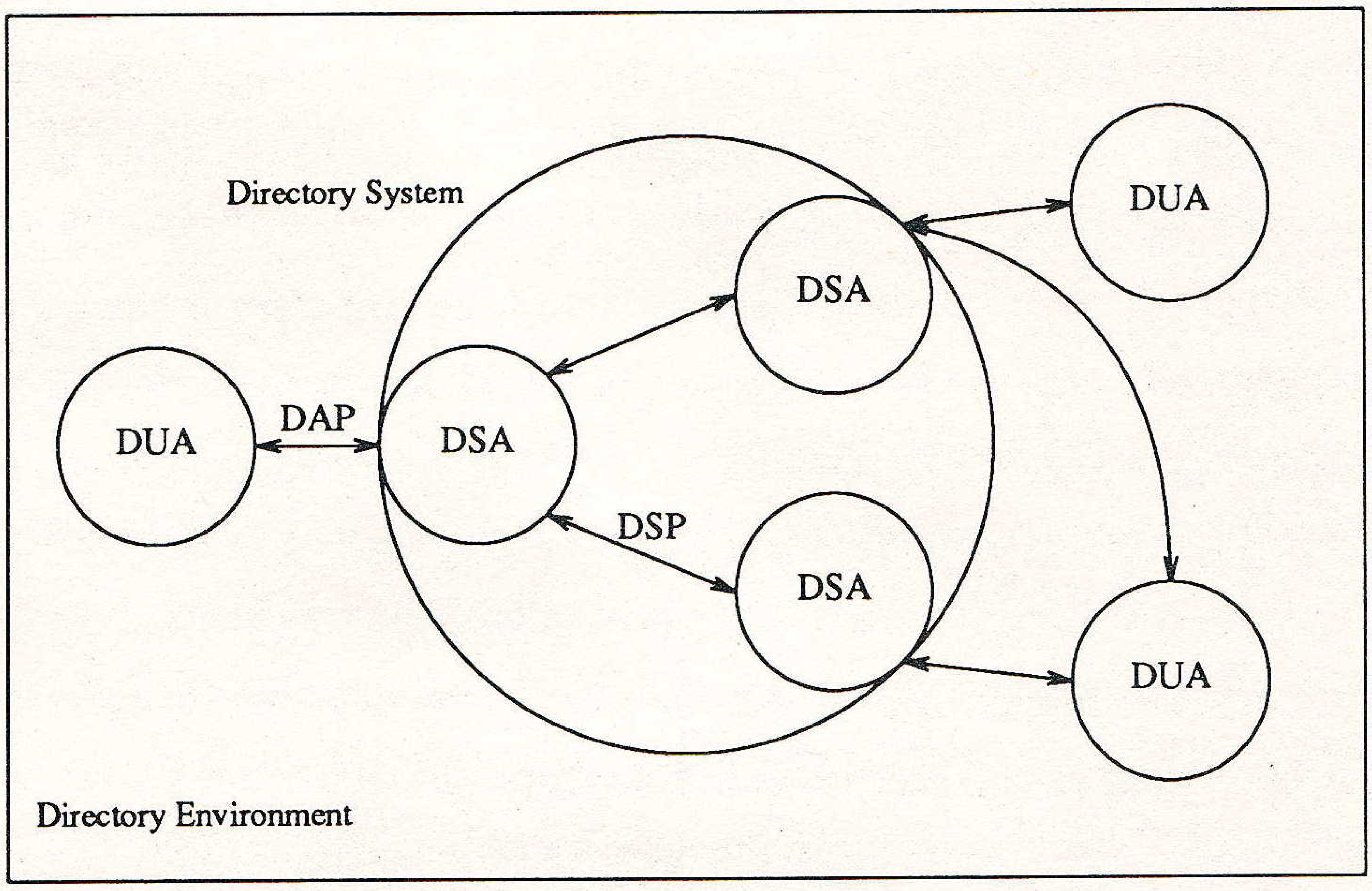


Figure 3: DSA—DUA relationship

The provision of the Directory service by DUA and DSA functional entities is shown in Figure 3. DSAs may utilise several modes of interaction from the recursive "chaining" of operations to the more iterative "referral" mode. The X.500 standard provides a detailed description of the distributed operation of the Directory, including the protocol supporting interaction between DSAs. This is called the *Directory System Protocol* and allows different DSA implementations, under different managements, to cooperate to provide a Global Directory Service.

The Directory Access Protocol and Directory System Protocol are OSI application layer protocols. Thus, the Directory resides in the upper layer of the 7 layer OSI model, with DSAs and DUAs defined as OSI application processes. Furthermore, it is intended that Directory protocols will be supported by the OSI "Remote Operations Service."

Authentication

The OSI Directory standard defines the use of the Directory for the management of "credentials" for authentication purposes. Support is provided for both weak authentication, via the storage of passwords, and strong authentication via the storage of "certificates" and "encryption keys." Authentication is a somewhat orthogonal issue and is not considered further by this article.

Summary

The OSI Directory standard specifies a directory service to support present and future users and applications within the expanding OSI world. The provision of a globally unified namespace is the cornerstone of this support and, in this respect, Directories fulfil the role of existing nameservers. In fact, there are many similarities between X.500 and services such as the Internet DNS. Hierarchical name spaces, and the functional model are two examples.

However, it is important to understand that the role of Directory Services extends beyond that of nameservers and that, in general, they will have a wider scope. Provision of a human usable information service is one aspect of this.

The broader scope of X.500 leads to the following notable features:

- The Directory will support a wide range of attributes. Hence, attributes are typed.
- The wide range of attributes and entries implies the need for schemas controlling the structure of information. Schema management is an issue for the next standardisation period.
- The Directory supports complex browsing and searching facilities.

The future

The OSI Directory standard became available during late 1988 and a number of implementations are already under development. There is a clear need for further implementation work allowing experimentation and interconnection. Like many services, the Directory must reach critical mass before it will be truly useful. This should be achieved as soon as possible.

The 1988 version of the OSI Directory standard is not the end of the story. The standards bodies are entering another round of standardisation during which many urgent, outstanding issues must be resolved. These include access control, improved modify functionality, and schema and system management tools. However, the present standard should be sufficient to produce working implementations and fill the crucial gap within the OSI communications environment.

References

[1] ISO and CCITT, "Information Processing Systems—Open Systems Interconnection—The Directory," ISO 9594-1-8, CCITT X.500-X.521, 1988.

[2] Mockapetris, P., "Domain Names: Concepts and Facilities," RFC 1034, 1987.

[3] Oppen, Derek C. & Dalal, Yogen K., "The Clearinghouse: A Decentralised Agent for Locating Named Objects in a Distributed Environment," Xerox Office Products Division, 1981.

[4] ISO, "Information Processing—Open Systems Interconnection: Specification of Abstract Syntax Notation One (ASN.1)," International Standard 8824, 1986.

STEVE BENFORD received his BSc. from the University of Nottingham in 1985. For the past three years he has been studying Directory Services, culminating in his PhD thesis "Research into the Design of Distributed Directory Services" which was passed in January 1989. Other research interests include advanced group communication and Computer Supported Cooperative Work within the European AMIGO and COSMOS projects. Current interests include COSMOS and the development of network management services, particularly their relation to Directory Services.

Reader Survey

Attached to this issue of *ConneXions* is a reader survey. We want to ensure that this newsletter is a valuable resource for you. Please take a few moments to complete the questionnaire. Return it to us by July 15th, and your subscription will be extended one month. We look forward to receiving your comments.

The Merit/NSFNET Information Services

by Laura Kelleher, Merit Information Services

Introduction

One of the major criticisms of the national networking environment is the need for more development of user-support mechanisms and user-friendliness. The National Science Foundation in its Project Solicitation for the Management and Operation of the NSFNET Backbone Network recognized the critical need for Information Services. The Merit Computer Network's proposal to the National Science Foundation (NSF) included a provision for comprehensive information and technical support services. In November 1987, MERIT, Inc., a computer network consortium of eight state-supported universities in Michigan, entered into a cooperative agreement with NSF to re-engineer and manage an enhanced backbone network.

Components

Merit's innovative Information Services components for managing the NSFNET backbone consists of two principal parts: an *Information Center*, with the responsibility for information dissemination, information resource management, and electronic communications; and a *Technical Support Group*, with the responsibility for providing direct staff-to-staff support for site managers, liaisons, and other staff at the mid-level and campus-level sites.

Objectives

Merit's plan to the National Science Foundation outlined the following objectives for Information Services:

- Provide an integrated system of information inquiry, retrieval, reporting, and management that allows easy user access from anywhere on the network.

- Provide a full range of related services such as tutorials, seminars, trouble-shooting, training, and paper and online documentation.

- Provide immediate availability of applications such as electronic mail, computer conferencing, electronic publishing, and database access.

The focus of Merit's NSFNET Information Services (IS) is direct technical support for mid-level sites and online systems which are accessible by the broader technical and user communities. IS has developed a number of mechanisms for distributing information about the NSFNET backbone. This includes the implementation of online systems as well as a number of seminars and publications. Merit is fortunate to have both Network Information Center and Network Operations Center facilities and services in the same location. The current IS staff is divided into three functional groups to provide these services: *Communications*, *Online Systems*, and *Site Liaisons*. At present, the IS staff consists of a manager and nine full time staff members. Two additional employees will provide Information Services to the CICnet community.

NSFNET background

Merit is undertaking the NSFNET effort in partnership with IBM and MCI, along with major financial support from the Michigan Strategic Fund. IBM is providing the packet-switching and network-management equipment and software, and MCI is providing the long distance transport facilities.

Dear ConneXions Subscriber:
We want to be sure that *ConneXions–The Interoperability Report™* is a valuable resource for you. Please return this questionnaire to us by July 15 and receive a **one month extension to your ConneXions subscription.** We look forward to receiving your comments.

Name ____________________

Company ____________________

Address ____________________

City ____________________ State ____________________ Zip ____________________

Phone ____________________

1. *How long have you been a ConneXions subscriber?*

☐ Less than one year ☐ One to two years ☐ More than two years

2. *Which of the following most closely describes your job function?*

☐ Engineering
☐ Research & Development
☐ Network Management
☐ MIS/Systems Integrator
☐ Operations Management
☐ Corporate Management
☐ Other (please describe) ____________________

3. *What is your opinion of the technical content of ConneXions?*

☐ Content is appropriate
☐ Not technical enough. I would like to see the following technical areas covered: ____________________

☐ Additional areas of interest I would like to see addressed: ____________________

The article that I found to be the most valuable was ____________________

4. *We would like your input on the format. Are you able to read ConneXions from cover to cover?*

☐ Yes ☐ No If No, why not? ____________________

Are you able to locate the articles of interest easily?

☐ Yes ☐ No If No, why not? ____________________

5. *Will you be renewing your subscription to ConneXions?*

☐ Yes ☐ No If No, why not? ____________________

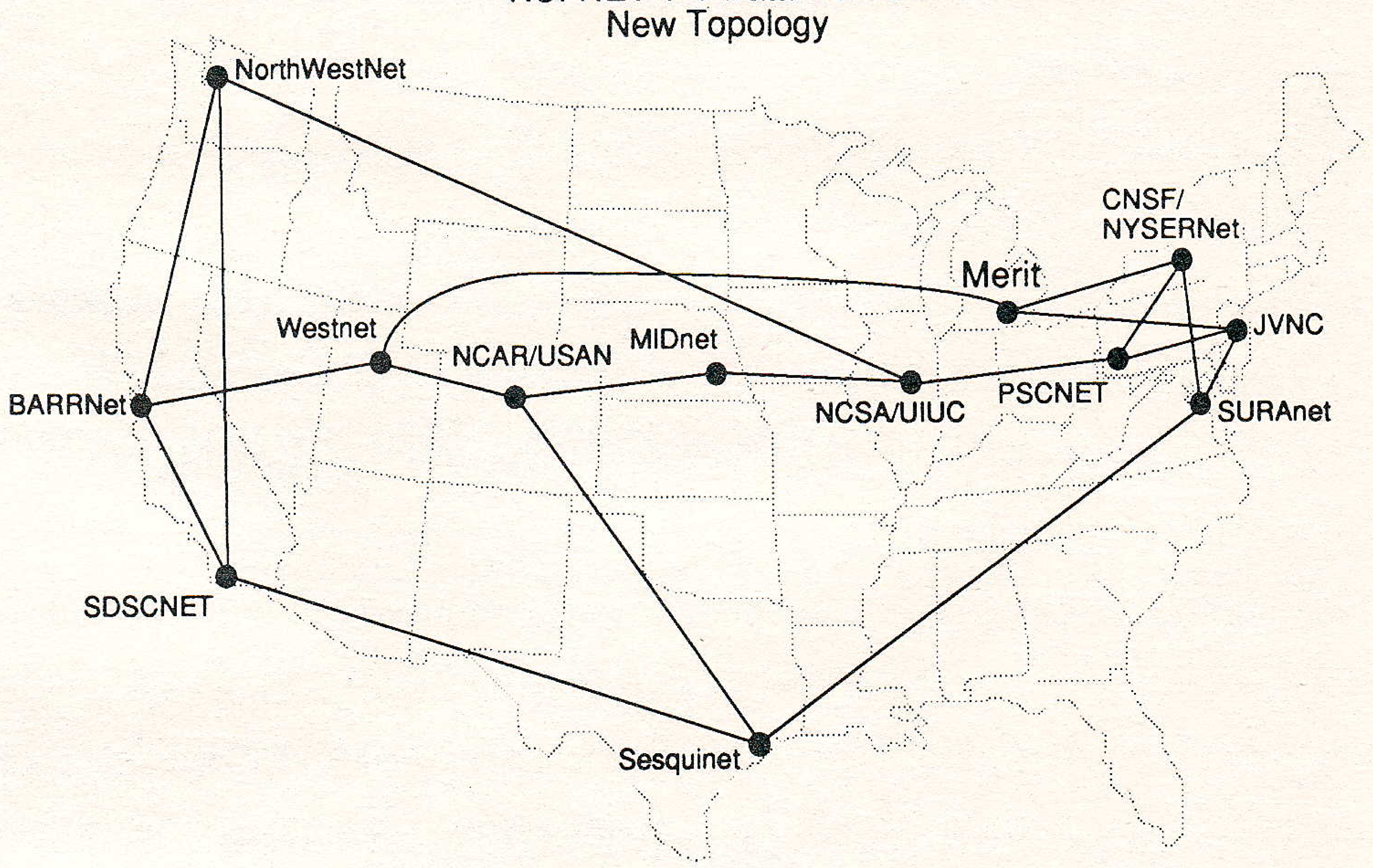
The experience and technology of IBM and MCI, along with the participation of the NSFNET midlevel networks, played a critical role in achieving on-schedule operational status for the re-engineered NSFNET backbone in July 1988.

Merit's plan specifically addressed improvements in connectivity and availability, improvements that would require carefully centralized management of network resources and the use of around-the-clock network operations staff. The re-engineered network is based on T-1 (1.544 Mbps) circuits linking thirteen regional nodes. As of early March, over 400 campuses, government agencies and research institutions have connections through the thirteen regional networks. The network is carrying over 160 million packets per week, making it possible for researchers to collaborate easily even when they are geographically distant.

Upgrades

An article in *ConneXions*, Volume 2, No. 12, December 1988, detailed the Merit implementation of the new NSFNET. The map below outlines a redesign to the NSFNET topology which is part of planned architecture improvements. The topology change will increase the number of T-1 circuits in the backbone to provide multiple connections for all nodes and take advantage of MCI's *Digital Reconfiguration Service* (DRS) to improve network management capabilities. The new topology is scheduled for implementation starting second quarter 1989.

NSFNET T-1 Data Network
New Topology



The Merit Computer Network

Electronic delivery of information

A critical resource in developing and deploying electronic information is an IBM-supplied 4381 mainframe dedicated to Information Services. The Network Information Services machine, NIS.NSF.NET or 35.1.1.48, is located at the Merit Network Operations Center on the University of Michigan campus. Services available on this machine include files available for File Transfer Protocol (FTP) and an electronic mail query system which is based on Remote SPIRES™ (*Stanford Public Information Retrieval System*).

continued on next page

Merit/NSFNET Information Services *(continued)*

In general, FTP allows users to communicate with a remote host and transfer both text and binary files between a local machine and a remote machine. The Remote SPIRES system enables users to send messages to a server which responds with information from its databases.

A number of databases can now be reached through such servers. The server can be queried by sending a message to:

nis-info@nis.nsf.net or nis-info@merit on BITNET.

If assistance is needed in using the server, a message should be sent to the Information Services staff at the following address:

UserHelp@nis.nsf.net or UserHelp@merit on BITNET.

Commands

Commands for the server should be the first text line of the message. Additional lines of the message will be ignored. In response to the command HELP, the server will return a list of the other commands that are available. Server commands currently available on the IS machine include:

INDEX: provides indices of directories currently available.

SEND: allows for retrieval of documents, especially useful for BITNET sites.

CONTACTS: A list of user, administrative, and technical contacts for a specific site.

TOPIC: this command allows for the retrieval for NSFNET network documents or Link Letter articles on a specific topic.

TOPOLOGY: Provides information regarding the current NSFNET topology.

Files

The document directories that are available on the NSFNET-IS machine for anonymous FTP include:

IEN: the SRI-NIC collection of Internet Engineering Notes.

LINKLTTR: all issues of the Merit/NSFNET newsletter the Link Letter.

NDOC: a repository for NSFNET related documents

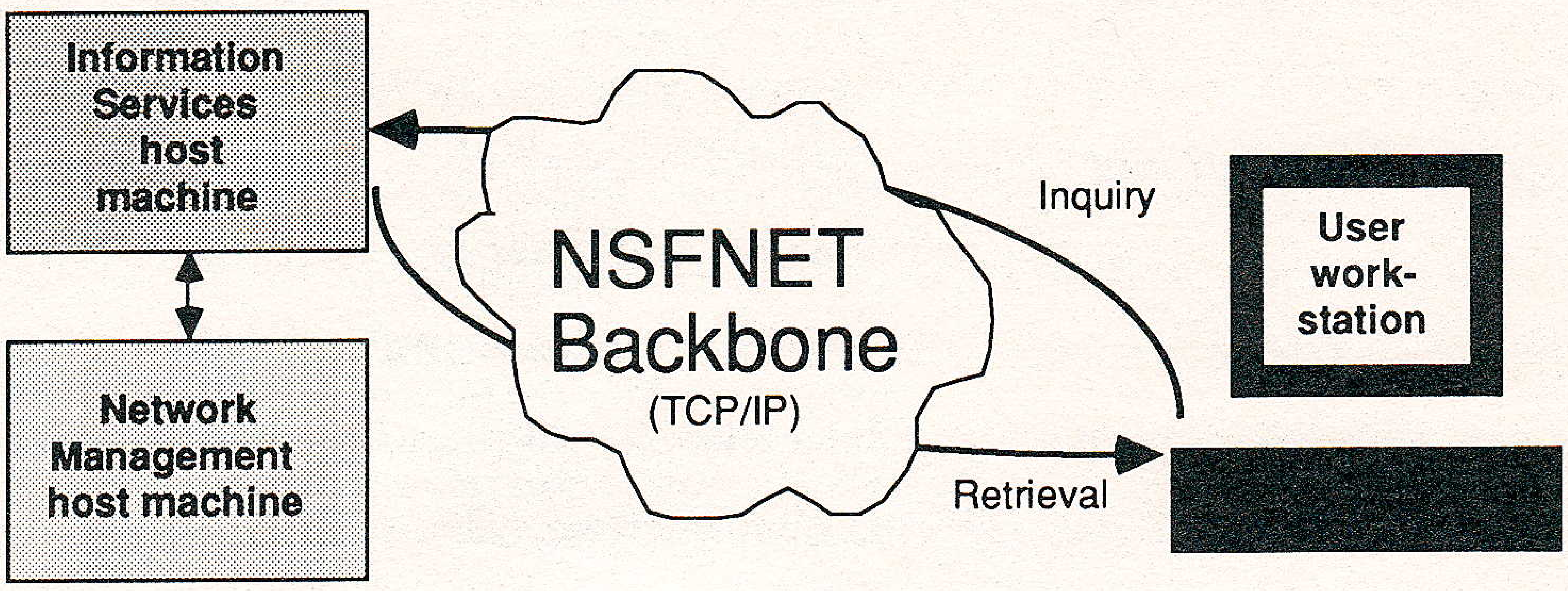
NSFNET: Merit-NSFNET "administrative" files

RFC: all currently available Requests for Comments from the SRI-NIC.

The availability of files for anonymous FTP and the mail-based query system are helping to meet the information needs of the NSFNET user community.

GRASP

Merit has begun work on a vehicle which will ultimately handle all aspects of electronic distribution of information for the Merit-NSFNET team and the NSFNET community—*GRASP*. GRASP is a combination of an IBM front-end system (GRANDiose Distributed Application System or GRAND) and a powerful database management system back-end, SPIRES. Together they will provide both standard mechanisms as well as fully-sophisticated functionality to service users. This system, when fully implemented, will move electronic information delivery into a more friendly domain.



Informational documents

As part of Information Services effort to meet the documentation and information needs of the NSFNET community, the Link Letter, a bi-weekly newsletter in hard copy and electronic versions, was created in April 1988. As the need for longer more in-depth articles arose, the Link Letter was expanded to a multipage format and a monthly publication schedule. To receive the electronic version, send a message to NSFNET-Linkletter-Request@merit.edu.

In accordance with the agreement to NSF, Information Services also prepares monthly, quarterly, and annual reports of the activities and statistical analysis of the NSFNET backbone project. Monthly and annual reports are available to interested researchers and technical personnel. These reports include packet counts, summaries of the number of networks attached to the backbone, packet delay information, comparisons between the amount of traffic carried on the old backbone versus the amount carried on the new backbone, and network monitoring improvements. Data collected from each node is processed and stored in a SPIRES database for easy reporting and examination of network performance.

Seminars

As part of its training and public relations efforts, Information Services in concert with the National Science Foundation has embarked on an effort geared to make the transition to intercampus networking less painful for those institutions new to internetworking. Regional workshops will bring the expertise of NSF, Merit, and other networking experts to groups of people identified by their campuses as the "liaisons" for internetworking. These two day sessions will give these representatives the information and tools they need to aid their campuses with internetworking. Information Services is planning on offering these seminars on a regular basis.

continued on next page

Merit/NSFNET Information Services *(continued)*

Other training and informational sessions have been presented by Merit for the technical and administrative personnel at the NSF regional sites. Talks and seminars for user groups and other professional organizations are helping researchers understand and access NSFNET's current and future capabilities. For more information about these seminars or other Merit/NSFNET services, send a message to NSFNET-INFO@merit.edu or call 1-800-66-MERIT.

While Merit has made a commitment to improve the speed and the functionality of the NSFNET backbone, it is also dedicated to streamlining methods of providing information and technical support services to insure that the increased speeds and services will directly benefit users. Merit is committed to developing improved Information Services which will promote a more user-friendly and user-accessible network.

LAURA A. KELLEHER is a technical writer and consultant for the NSFNET project in Merit's Information Services. Laura previously worked as Senior Instructor for the Residence Halls Computer Program (RESCOMP), a joint effort between the Housing Division and the Computing Center to promote student computer usage at the University of Michigan.

Upcoming Events

Tutorials

There is still time to sign up for Advanced Computing Environments' *TCP/IP OSI Internetworking Tutorials* which will be offered in Dallas, June 19–22, 1989. Call 415-941-3399 for more information.

INTEROP 89

Plans are well underway for the next INTEROP™ conference and exhibition which will be held in San Jose, CA, October 2–6, 1989. The format is two days of tutorials (17 in all) followed by 3 days of technical conference sessions. There are a total of 35 technical sessions in 5 parallel tracks. Additionally, Birds Of a Feather (BOF) sessions will allow attendees to discuss topics of common interest in an informal atmosphere. We have already scheduled a number of BOFs, but welcome your suggestions for topics which may be suitable for such sessions.

Exhibition

Concurrent with the conference is the INTEROP exhibition where over 100 vendors will be connected to a "Show and Tel-net" and demonstrate interoperable systems based on TCP/IP and OSI. The shownet will be connected to the worldwide TCP/IP Internet, and attendees will be able access their Internet mailbox from several electronic mail centers in the convention center area.

Of special interest this year are the emerging X Windows systems and early OSI products. Several collaborative demonstrations of such technologies will be on display at INTEROP 89.

A detailed advance program will be mailed to you in early July.

Network Management Update

On April 18, 1989, The Internet Activities Board (IAB) issued two new RFCs which pertain to network management in the Internet. (See *ConneXions*, Volume 3, No. 3, March 1989). The two RFCs are 1095 and 1098.

CMOT

RFC 1095 is entitled *The Common Management Information Services and Protocol over TCP/IP (CMOT)* and defines a network management architecture that uses the International Organization for Standardization's (ISO) Common Management Information Services/Common Management Information Protocol (CMIS/CMIP) in a TCP/IP environment. This architecture provides a means by which control and monitoring information can be exchanged between a manager and a remote network element. In particular, this memo defines the means for implementing the Draft International Standard (DIS) version of CMIS/CMIP on top of Internet transport protocols for the purpose of carrying management information defined in the Internet-standard management information base. DIS CMIS/CMIP is suitable for deployment in TCP/IP networks while CMIS/CMIP moves toward becoming an International Standard. Together with the relevant ISO standards and the companion RFCs that describe the initial structure of management information and management information base, these documents provide the basis for a comprehensive architecture and system for managing TCP/IP-based internets, and in particular the Internet.

SNMP

RFC 1098 is called *A Simple Network Management Protocol (SNMP)* and is a re-release of RFC 1067, with a changed "Status of this Memo" section. This memo defines a simple protocol by which management information for a network element may be inspected or altered by logically remote users. In particular, together with its companion memos which describe the structure of management information along with the initial management information base, these documents provide a simple, workable architecture and system for managing TCP/IP-based internets and in particular the Internet.

IAB Policy

The IAB has designated these two different network management protocols with the same status of "Draft Standard" and "Recommended."

The IAB intends each of these two protocols to receive the attention of implementers and experimenters. The IAB seeks reports of experience with these two protocols from system builders and users.

By this action, the IAB recommends that all IP and TCP implementations be network manageable (e.g., implement the Internet MIB) and that the implementations that are network manageable are expected to adopt and implement at least one of these two Internet Draft Standards.

Getting RFCs

RFCs can be obtained via FTP from SRI-NIC.ARPA with the pathname RFC:RFCnnnn.TXT where "nnnn" refers to the number of the RFC. Log in with FTP username ANONYMOUS and password GUEST. The NIC also provides an automatic mail service for those sites which cannot use FTP. Address the request to SERVICE@SRI-NIC.ARPA and in the subject field of the message indicate the RFC number, as in "Subject: RFC 1098." The NIC also sells hardcopy versions of the RFCs, call 415-859-3695 for more information. Submissions for RFCs should be sent to Postel@ISI.EDU.

Book Review

The Design and Implementation of the 4.3BSD UNIX Operating System by Samuel P. Leffler, Marshall Kirk McKusick, Michael J. Karels and John S. Quarterman, Addison-Wesley Publishing Company, Reading, Massachusetts, 1989, 471 pp. including glossary and index. ISBN 0-201-06196-1.

Background

With the current flood of "how-to" and "me-too" books in the computing field the appearance of a text with true authority is indeed a significant event. "The Design and Implementation of the 4.3BSD UNIX Operating System" was written by principal developers of this important version of UNIX. The 4.3BSD release of UNIX came out of the Computer Science Research Group, University of California at Berkeley in 1986. When people speak of a schism being repaired in the UNIX world it is primarily between those based on 4.3BSD (and its predecessor, 4.2) and others derived from AT&T's System V which are being merged by organizations like the Open Software Foundation and UNIX International.

New features of 4.2 and 4.3BSD were motivated by the desire of DARPA to make available a UNIX release which provided full support for the Internet protocol suite and other ideas from the research and commercial community. The first distribution incorporating these networking facilities appeared in 4.2BSD in 1983. The release three years later of 4.3BSD added many improvements and enhancements.

Organization

The book is organized as a tour through the kernel structure which makes up the heart of the system. The text is rich not only with descriptions of how the software works but the thoughts that went into its design and discussions of alternatives which were considered along the way. The pages are liberally sprinkled with comments describing problems they still see with their design and ways in which it might be improved in a future release. No one but the developers could provide such rich commentary, it is as if you had the opportunity to sit in a room with them and listen as they discussed the issues.

Style

The style of the book is interesting. Besides being broken up into major and minor sections each topic is presented in well-chosen bite-size pieces about a page or so in length. This neatly allows you to read a complete topic, study it and go onto the next without feeling overwhelmed. At the end of each chapter are exercises which don't simply re-hash the factual material but often challenge the reader to propose alternate designs to those described. You are drawn into the fast-paced technological atmosphere in which the software grew.

The book is divided into five parts, Overview, Processes, I/O System, Interprocess Communication and System Operation. At the end is an extensive glossary and index. Every chapter is rich with references to other work influential to the system's design.

IPC

Of particular interest is the section on *Interprocess Communication* (IPC). It will enrich any previous study of the subject of networking as it goes beyond mere protocol descriptions and describes in great detail the challenges faced when realizing a networking standard in the context of a general purpose operating system.

One topic which comes through clearly in this section is that it is not enough to merely implement protocols as described in the RFCs. Major problems must be solved such as those posed by the memory management of packets and presenting operational access to multiple hardware interfaces and routing environments. The following is from their description of how they arrived at the current socket interface:

"For several reasons, binding a name to a socket was separated from creating a socket. First, sockets are potentially useful without names. If all sockets had to be named, users would be forced to devise meaningless names without reason. Second, in some communications domains, it may be necessary to supply additional, nonstandard information to the system before binding a name to a socket—for example, the "type of service" required when using a socket. If a socket's name had to be specified at the time the socket was created, supplying this information would not be possible without further complicating the interface." (page 285)

"The interface to the interprocess-communication facilities was purposely designed to be orthogonal to the standard UNIX system interfaces—that is, to the *open*, *read*, and *write* system calls. This decision was made to avoid overloading the familiar interface with undue complexity. In addition, the developers thought that using an interface that was completely independent of the UNIX filesystem would improve the portability of software because, for example, UNIX pathnames would not be involved. Backward compatibility, for the sake of naive processes, was still deemed important; thus the familiar read—write interface was augmented to permit access to the new communications facilities wherever it made sense (e.g., when connected stream sockets were used.)" (page 287)

Development history

The detailed descriptions of how performance was improved over time provides great insight into just how difficult it is to go from a networking specification to an implementation. The current version of the 4.3BSD TCP code achieves outstanding throughput and robustness in the face of congestion and other network stress. How the algorithms evolved to this level of sophistication is described with a balanced treatment of both theoretical and practical issues.

After an extensive discussion of how the Internet protocols were provided there is a description of the XNS implementation within the kernel. This provides a useful illustration of how multiple network protocols can exist within one operating system all using a common programming interface. They describe their intentions of providing a full implementation of the OSI protocol stack in a future release using the same model.

This text is not meant to be an introduction to the subject of operating systems. The level of the material is suitable as a companion text in an operating systems course along with a general-purpose textbook, or for anyone who has a good background in the subject. Having taught such courses I would certainly recommend it to educators as one of the rare opportunities to provide a concrete, implementation oriented description of the ideas they are studying. It provides the palpability so often lacking in upper-level computer science courses. Anyone reading this book will find it a source for specifics about 4.3BSD and more general systems problem solving examples.

—*Barry Shein*

Department of Defense High Level Protocols Conformance Testing Program

by Martin R. Gross,
Defense Communications Engineering Center

In an effort to ensure compliance with its Military Standard High Level Data Communications Protocols (IP, TCP, FTP, SMTP, Telnet) and increase the probability of interoperability in its diverse multi-vendor environment, the Department of Defense (DoD) has initiated a program to certify vendor implementations of these products. As Executive Agent for the DoD Data Communications Protocols, the Defense Communications Agency (DCA) has been tasked with implementing this program.

Background

The policy for high level protocol conformance testing was established in a memorandum from the Assistant Secretary of Defense for Command, Control, Communications and Intelligence dated August 26, 1988. The memorandum mandates conformance testing on all new contracts executed after June 1, 1989. Products procured under contract must be tested by a National Institute of Standards and Technology (NIST) accredited laboratory prior to operational use on any DoD network. The memorandum also establishes a *Qualified Products List* which will be maintained by DCA. For a product to be placed on the Qualified Products List, acceptable tests results must be presented to DCA from an accredited laboratory which is independent of the vendor.

NVLAP

DCA started an in-house testing program for DDN X.25 in 1983. Due to limited resources however, this program could not be continued in-house nor could a high level protocol test program be developed in the same manner. For this reason DCA turned to the *National Voluntary Laboratory Accreditation Program* (NVLAP) run by NIST. Under NVLAP, laboratories are recognized and accredited to perform specific testing services aimed at evaluating products to determine if they meet applicable standards. As the program's name specifies, this is a *voluntary* program and NVLAP accreditation does not imply the certification of products or test data. In July 1988, DCA requested that NIST establish a NVLAP for the DoD Protocols (DDN X.25 and the five High Level Protocols). Formal establishment of the program was announced in the *Federal Register* on July 21, 1988. The X.25 Program was established first and there are currently three accredited laboratories. The NVLAP for the High Level Protocols is now being developed.

Handbook

The NVLAP Handbook for the High Level Protocols which presents the operational and technical requirements for an accredited laboratory was published in draft form on March 22, 1989. The document was mailed to all those who replied to the Federal Register announcement and was open to public comment until the 14th of April. Laboratory applications are now being accepted by NIST and the first laboratory will be accredited by the middle of June. It is expected that there will be a minimum of three accredited laboratories.

The laboratories will be required to use the *DCA Upper Level Protocol Test System* to perform the testing service. The system was developed under DCA contract to provide a standard testing capability for the DoD High Level Protocols.

The system tests protocol functionality including; upper layer interfaces, validity of outputs, and input error handling. (See *ConneXions* Volume 2, No. 8, August 1988 for further details). The test system operates on a VAX™ CPU running Ultrix™ 1.1 and is publicly available from the National Technical Information Service. The system has been in use since December of 1987 and is currently in use by ten organizations for in-house testing.

Testing Circular

To clarify testing policies and provide guidance to vendors, DCA will publish a *DoD High Level Protocols Testing Circular*. The circular will establish specific testing policies relating to the testing of products across hardware lines and the retesting of modified products. The circular will also establish the procedure for placement of products on the Qualified Products List. The DoD Protocol Conformance Testing Profile will also be included in the circular. This Profile establishes the set of mandatory features that must be implemented in each protocol. It also indicates those features which are optional. This Profile has been approved by the DoD's Protocol Standards Steering Group but is still available from DCA for public comment. The testing circular will be published in draft form by June 1.

Test system

To help vendors prepare their products for laboratory certification, DCA has installed a test system that can be used by vendors. This system will be available for the next nine months on a first-come first-served basis. The system is accessible through the Internet or a dial-up link and is available for self testing with no online support. Information is provided below on how to obtain further information on this program.

Comments/questions relating to protocol testing can be addressed to:

Martin Gross,
DCA Code R640
1860 Wiehle Avenue
Reston, VA 22090-5500
Email: martin@edn-unix.dca.mil or martin@protolaba.dca.mil.

For NVLAP information or documents contact:

Jeff Horlick
NVLAP
National Institute of Standards and Technology, Bldg 411
Gaithersburg, MD 20899
301-975-4016.

For the DCA Upper Level Protocol Test System and Documents contact:

National Technical Information Service
Springfield, VA 22161
703-487-4807. (Product #AD-A204-558).

VAX and Ultrix are Trademarks of Digital Equipment Corporation.

MARTIN R. GROSS is an Electronics Engineer at the Defense Communications Engineering Center. He has been involved with the DCA Upper Level Protocol Test System since 1986 and is currently responsible for the implementation of the DoD's High Level Protocols Testing Program. Martin received his B.S. in Electrical Engineering from Drexel University and will receive an M.S. in Electrical Engineering from Virginia Tech in December.

CONNEXIONS
480 San Antonio Road
Suite 100
Mountain View, CA 94040
415-941-3399
FAX: 415-949-1779

FIRST CLASS MAIL
U.S. POSTAGE
PAID
SAN JOSE, CA
PERMIT NO. 1